UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería de Tecnologías y Servicios de Telecomunicación

TRABAJO FIN DE GRADO

AUTO-CALIBRACIÓN DE REDES DE CÁMARAS

Carlos Fernández Herrero
Tutor: Marcos Escudero Viñolo
Ponente: Jesús Bescós Cano

Junio 2019

AUTO-CALIBRACIÓN DE REDES DE CÁMARAS

Carlos Fernández Herrero
Tutor: Marcos Escudero Viñolo
Ponente: Jesús Bescós Cano

Video Processing and Understanding Lab
Departamento de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio 2019

Resumen

El presente Trabajo de Fin de Grado tiene como objetivo la implementación de un método de auto calibración de cámaras. El desarrollo de este trabajo se ha basado en la publicación titulada "Automating Multi-Camera Self-Calibration", [1], del que se ha extraído la información básica. El método propone una situación en la que se utiliza un proyector para proyectar un patrón espacio temporal de luz sobre un conjunto de objetos estáticos. Este patrón sirve para añadir textura a la escena y permite, en teoría, facilitar la búsqueda de puntos de interés que faciliten la calibración extrínseca de la escena. Este proyecto estudia la forma en la que se diseña y desarrolla dicho algoritmo haciendo referencia a temas como la geometría proyectiva, matrices fundamental y de proyección, detección, descripción y búsqueda de correspondencias entre imágenes, la calibración intrínseca y extrínseca de las cámaras y métodos de corrección de errores como *Bundle Adjustment*.

Se ha desarrollado un algoritmo que tiene en cuenta la información espacial y temporal de la proyección de un patrón determinado para caracterizar las regiones proyectadas por el patrón de luz.. Dicho proceso finaliza en la calibración extrínseca de las cámaras involucradas en el proceso de grabación.

El trabajo finaliza con un estudio del rendimiento del método y del impacto de los parámetros en la calibración obtenida. Cuantitativamente, se estudian los errores de retro-proyección obtenidos tras la calibración. Cualitativamente se estudia la proyección 3D del patrón.

Palabras clave

Geometría proyectiva, matriz fundamental, calibración por proyección de patrones de luz, reconstrucción 3D.

Abstract

The objective of this Final Degree Thesis is the development of an auto-calibration camera algorithm. This project is based on the paper 'Automating Multi-Camera Self-Calibration' which has been used to extract the main information from it .We propose here a scene where we use a projector to create several projection patterns over a set of static objects. This method is completely automatic and allows using any kind of pattern to perform the calibration and to be robust to errors. We study about how to design and develop the algorithm including topics like geometry projection, fundamental matrix and projection matrix, detection, description and image matching, intrinsic and extrinsic calibration and error correction methods like *Bundle Adjustment*.

The algorithm has been developed taking in consideration spacial and temporal information of the projection of a defined pattern to be robust to errors. The process ends with the intrinsic and extrinsic calibration of all involved cameras.

Finally, we study the perfomance of the method and the impact of the calibration parameters obtained. Moreover, we study reprojection erros and the 3D projection of the pattern.

Keywords

Projective geometry, fundamental matrix, calibration by projecting light patterns, 3D reconstruction.

Agradecimientos

A mi abuelo, Vicente.

Estos años han sido increíbles, muchísimas experiencias nuevas, mucha gente conocida, muchas horas empleadas en una pasión. Haya costado más o menos por fin esta terminado. Una etapa mas de la vida terminada. Eso si, ojalá me hubiera visto mi abuelo defender este trabajo, que mejor o peor, es algo mio, y seguro que le encantaría ver a su nieto cumpliendo una etapa mas de su vida. Te quiero.

Simplemente gracias, gracias a todas aquellas personas que me han apoyado, mi familia, mis amigos...todos. En especial a una de las personas mas especiales que he conocido nunca: Irene, mi guía en todo esto de dejar las cosas bonitas, la organización, la previsión, tener todo siempre controlado...Su constancia y sus ganas de trabajar me parecen increíbles.

No me voy a extender mucho más. A por la siguiente etapa de la vida. Gracias a todos aquellos que me apoyaron en algun momento. Lo aprecio mucho.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

1.1. Motivación

La calibración de una escena es el proceso de obtención de los parámetros que definen el proceso de transformación del mundo 3D a los planos 2D de las imágenes donde queda capturado. En escenarios capturados mediante múltiples cámaras (escenarios multi-cámara) el proceso de calibración tiene dos partes: la calibración intrínseca y la calibración extrínseca. La calibración intrínseca describe cómo se proyecta la escena en cada cámara. La calibración extrínseca relaciona las proyecciones de las diferentes cámaras. Esta última permite reconstruir cualquier punto de la escena si conocemos su posición en (al menos) dos de los planos de imágenes capturados. La reconstrucción será en principio mejor (más precisa) cuantas más correspondencias entre cámaras conozcamos para ese punto. Si incluimos además la calibración intrínseca de cada cámara, la reconstrucción será euclidiana, es decir, las relaciones entre los objetos de la escena serán proporcionales (iguales salvo por un factor de escala) a las del mundo 3D proyectado.

Para calibrar completamente, se necesita un conjunto (generalmente pequeño) de correspondencias entre puntos, es decir, definir las posiciones 2D del mismo punto en las imágenes de las cámaras. Al ser un proceso propenso a errores, cuantas más correspondencias correctas se utilicen mejor será la calibración.

El proceso de obtención de estas correspondencias es complejo y generalmente requiere del uso de un patrón de calibración o damero. Como alternativa, pueden usarse esquemas automáticos de búsqueda de correspondencias, que generalmente siguen el mismo patrón: descripción y detección de puntos de interés en cada cámara, búsqueda de correspondencias, estimación de la matriz fundamental y estimación de las matrices de proyección de cada cámara.

Estos esquemas presentan dificultades, por ejemplo, en las zonas planas u homogéneas de la imagen (problemas de detección) o en situaciones de alta distorsión proyectiva (problemas de descripción).

En este trabajo exploramos una solución distinta para la obtención de estas correspondencias, la proyección de un patrón texturado sencillo cuya caracterización es temporal, es decir, cuya apariencia varía con el tiempo. La aproximación presentada, basada en un trabajo de investigación, [1], puede entenderse como un ejemplo sencillo del tipo de información utilizado por algunos esquemas de estimación de la profundidad (e.g. Kinect [4]).

1.2. Objetivos

En este trabajo expondremos un estudio previo y el proceso para la reconstrucción 3D de la escena. Podemos ver el detalle de cada objetivo a continuación:

1. Comprender y estudiar el estado del arte en las áreas de geometría proyectiva, detección y descripción de puntos de interés, obtención de correspondencias y los métodos de segmentación y corrección de errores y reconstrucción 3D.

2. Diseño y desarrollo del trabajo propuesto. Estudio de todas las etapas que permiten comprender y realizar el método propuesto desde que capturamos la escena con cámaras hasta la extracción de las matrices fundamentales, de proyección y esenciales.

3. Evaluación de los resultados. Estudio de la viabilidad del método propuesto.

1.3. Organización de la memoria

La memoria consta de los siguientes capítulos:

- Capítulo 1. Motivación, objetivos del trabajo y organización de la memoria.

- Capítulo 2. Estado del arte. Estudio de la base teórica del trabajo. Geometría proyectiva, detección y descripción de puntos y obtención de correspondencias. Reconstrucción 3D.

- Capítulo 3. Diseño y Desarrollo. Proceso detallado para la calibración de una red de cámaras desde la captura de una secuencia de imágenes pasando por la asociación temporal y espacial hasta la calibración de las cámaras involucradas.

- Capítulo 4. Resultados experimentales. Comparación entre los algoritmos utilizados y datos cualitativos y cuantitativos de los resultados obtenidos..
- Capítulo 5. Conclusiones y trabajo futuro.
- Bibliografía.

Capítulo 2

Estado del arte

2.1. Introducción

Comenzaremos este capitulo proporcionando una visión general del trabajo realizado anteriormente y los fundamentos en los que nos hemos basado para realizarlo. Presentaremos los fundamentos teóricos de este trabajo y la forma en que obtendremos las distintas matrices necesarias para la reconstrucción 3D de una escena. Trataremos temas como la geometría proyectiva, el modelo de cámara *pinhole*, las distintas matrices de características que obtendremos y métodos de segmentación de regiones como *Ultrametric Contour Map* entre otros.

2.2. Concepto inicial

El trabajo se centra en la información publicada en [1]. En él se demuestra un método preciso de auto calibración de cámaras. Se debe usar para ello al menos dos cámaras, que no necesitan estar sincronizadas, y un proyector. Proyectando un patrón en predeterminado en el campo de visión de las cámaras se detectan puntos de interés (estarán definidos por una secuencia binaria). Así se resuelve el problema de correspondencias entre dos vistas adyacentes.

2.3. Geometría proyectiva

La geometría proyectiva permite trabajar con un modelo de captación de imágenes asumiendo las posibles distorsiones por sistemas físicos como las lentes. Estudia la relación entre figuras geométricas y su proyección en el plano. Para ello es necesario comprender el funcionamiento de los sistemas de captura y creación de imágenes

puesto que al capturar imágenes pasamos del mundo de tres dimensiones (3D) en el que vivimos al de dos dimensiones (2D) en el que se captura la imagen, con la consiguiente pérdida de información tridimensional.

Un concepto que interviene en esta transformación es el concepto del modelo de cámaras *pinhole* en el que se establece una relación matemática entre las coordenadas del punto en 3D del mundo real y su proyección en el plano de imagen. Dicha proyección depende de una combinación de parámetros extrínsecos e intrínsecos de la cámara contenidos en la denominada matriz de proyección, como se explicará en la sección 2.3.3. Todo lo anterior es básico para entender el concepto de geometría epipolar. Dicha geometría no depende del tipo de escena si no de los parámetros intrínsecos de las cámaras y de sus posiciones relativas. La geometría epipolar queda definida por la matriz fundamental, como veremos en la sección 2.3.4.

2.3.1. Modelo de cámara *pinhole*

Este modelo define la formación de imágenes en 2D. Se basa en dos fundamentos: dos puntos definen una recta y todo par de rectas se corta en algún punto (aunque sea en el infinito en el caso de rectas paralelas).

El modelo de cámara *pinhole*, expuesto en [5], se basa en la idea de una lente ideal (sin distorsión alguna) con una apertura que tiende a cero. Dicha apertura es atravesada por un haz de luz, que pasa por un centro óptico, consiguiendo así despreciar las posibles reflexiones y difracciones que las lentes producen.

El modelo define un centro óptico, C, en el cual la luz proyectada converge, y un plano de imagen, situado a una distancia focal, f, del centro óptico y perpendicular al eje óptico Z_c, donde se proyecta la imagen (ver Figura 2.1).

En [6], Faugeras explica como formar la imagen proyectada a través de transformaciones mediante los distintos sistemas de coordenadas representadas en la Figura 2.1.

- Sistema de coordenadas del mundo (X, Y, Z): describe la posición de un punto M (en el mundo real) respecto a un origen de coordenadas, que normalmente se establece como el centro óptico de la cámara, C.

- Sistema de coordenadas de la imagen (u, v): describe la posición de un punto m que representa la proyección de M en el plano imagen. El origen del sistema de coordenadas es c (punto en el que el eje óptico corta el plano de la imagen).

- Sistema de coordenadas de la cámara (X_c, Y_c, Z_c): coordenadas que definen la posición de un punto M respecto de la cámara. Tiene su origen en el centro óptico de la cámara, C. En este trabajo asumiremos que M y M_c son iguales.

- Sistema de coordenadas normalizadas de la imagen (u_n, v_n): definen la posición de un punto m_n en el plano imagen $I(u, v)$. Como en el caso anterior asumimos que m_n es igual a m. Al ser un sistema de coordenadas normalizado conseguimos que el origen de coordenadas sea la esquina superior izquierda del mismo plano, $c_n(u, v)$.

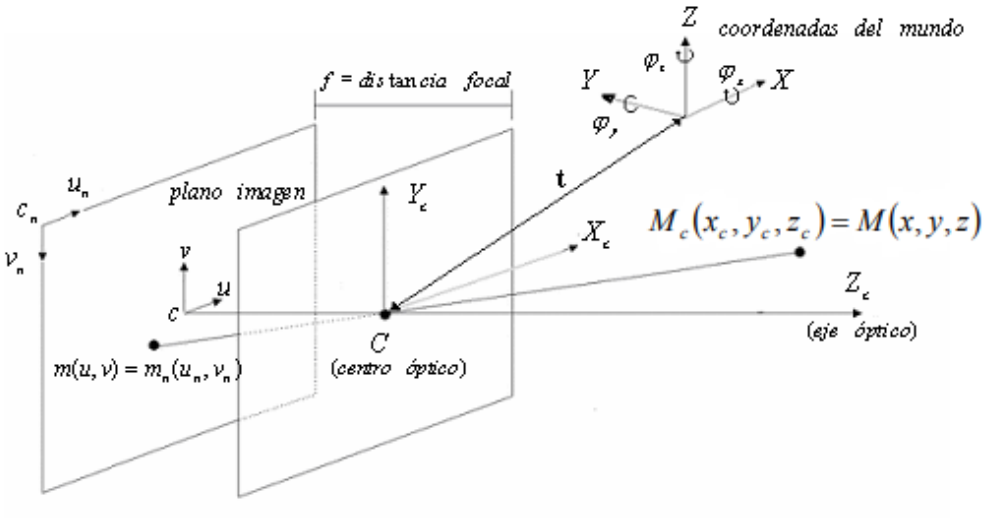

Figura 2.1: Modelo de cámara *pinhole,* extraído de [2].

Toda cámara tiene unos parámetros propios y otros relativos al lugar o posición en el que se encuentran. Estos parámetros son:

Parámetros extrínsecos

Información que describe como se relaciona la cámara y la posición relativa del objeto en el espacio, esto es, la traslación y la rotación. Se describen mediante el sistema de coordenadas del mundo.

1. Rotación: ángulo de rotación sobre cada uno de los ejes del sistema de coordenadas del mundo $(\varphi_x, \varphi_y, \varphi_z)$. Viene dado por la matriz de rotación, R.

2. Traslación: definido por el vector t, $(t_x, t_y, t_z)^T$, que mide la distancia entre el centro óptico de la cámara, C, y el origen de coordenadas del mundo (X, Y, Z).

Parámetros intrínsecos K

Usados para la definición de la geometría interna de la cámara. Tratan el proceso que se produce al ser atravesado la lente de una cámara por un haz de luz, el cuál

producirá la imagen final en 2D. Serán parámetros variables para cada cámara.

1. Centro óptico, c: formado por u_0 y v_0. Punto del plano en el que el eje óptico Z_c atraviesa el plano imagen. Al ser una imagen que forma parte del mundo real (2D) dicho punto se tratará en unidades de píxeles.

2. Factor de escalado, k_u y k_v : relación existente entre el tamaño del objeto real y el captado por la cámara. Dicha relación puede variar para cada sistema de coordenadas. El factor de escalado consta a su vez de tres parámetros:

   - Distancia focal, f: distancia entre el centro óptico, C, y el centro del plano imagen, c.
   - Factor de proporción, s: relación entre el tamaño de la componente vertical y la horizontal de un píxel en un sistema de coordenadas determinado.
   - Factor de conversión pixel-milimetros, d_u y d_v: proporción de píxeles por milímetro utilizados por la cámara. Relación entre píxeles de la imagen y tamaño de la imagen en milímetros (división de ambos términos).

Para el caso en el que no se introduce distorsión alguna, provocada por las lentes y elementos físicos de la cámara, los tres parámetros anteriores que componen el factor de escalado se relacionan multiplicándose entre si.

2.3.2. Geometría epipolar

La geometría epipolar entre dos imágenes es, como se describe en [3], la formada por la intersección del plano de la imagen con el conjunto de planos con origen en la *baseline* (tomando la línea que une ambos centros ópticos de las cámaras como referencia, en la Figura 2.2). Dicha geometría únicamente depende de los parámetros intrínsecos de las cámaras involucradas y de su posición relativa. Todas estas condiciones se pueden integrar en la conocida como matriz fundamental, descrita en la sección 2.3.4 de este capítulo.

Como se puede observar en la Figura 2.3a y según se describe en [7] y [3] entre otros, el punto M, que se proyecta como m y m', sobre los planos A y B respectivamente, está contenido en el mismo plano que ambos centros de proyección, C y C'. La línea de base o *baseline* une ambos centros de proyección y corta al plano de la imagen en los epipolos, e y e'. El plano que incluye dichos puntos se denomina plano epipolar, π.

Además, los planos que pasan por los centros ópticos de las cámaras y cualquier punto M, cortarán a los planos de la imagen en los epipolos. La geometría epipolar se basa en este hecho conocido como condición de coplanaridad.

El punto e' y m' forman parte de la linea epipolar, l', de la Figura 2.3. De esta relación se puede establecer que dado un punto M con proyección sobre el plano de la imagen m del plano A, el punto m' de la proyección sobre el plano B deberá estar contenido en la linea epipolar l'. Esta relación es útil para reducir las posibilidades a la hora de buscar correspondencias entre imágenes.

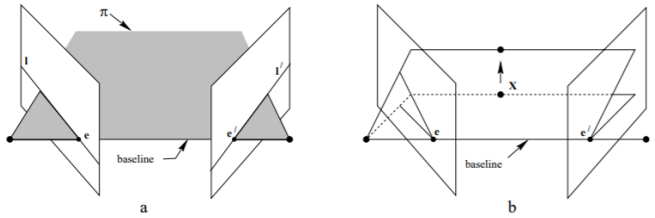

Figura 2.2: Geometría epipolar, extraído de [3].

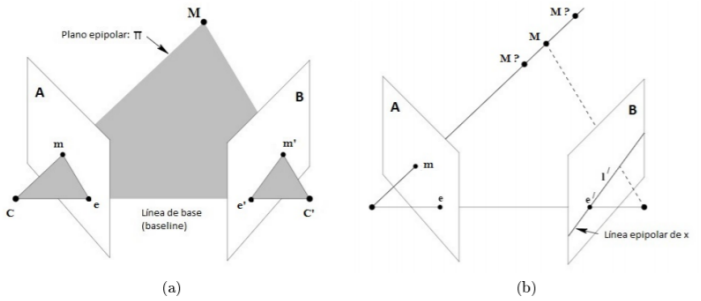

Figura 2.3: Puntos de correspondencia geométrica, extraído de [3].

2.3.3. Matriz de Proyección

Esta matriz de proyección, P, permite modelar las características de una cámara. Se trata de una matriz de dimensiones 3×4, la cual depende de la matriz de parámetros intrínsecos K y de los parámetros extrínsecos (R y t) expuestos en el anterior apartado, 2.1.

En el momento de captar la imagen debemos establecer la relación entre un punto, M, en el mundo real y la proyección sobre la imagen, m, (2D). Ambos parámetros se pueden relacionar mediante la siguiente fórmula, incorporando un parámetro de corrección, λ:

$$\lambda \cdot m = P \cdot M \tag{2.1}$$

En dicha ecuación, $m = (x, y)$, son las coordenadas en 2D de un punto dado. Expresándola en coordenadas homogéneas incorporamos un factor α ($\alpha = 1$ para el caso de coordenadas cartesianas) quedando de la siguiente forma: $m = (\alpha x, \alpha y, \alpha)$. Añadiremos un parámetro más para la coordenada en 3D, $M = (\beta X, \beta Y, \beta Z, \beta)$.

Transformación coordenadas del mundo (X, Y, Z) a coordenadas de la cámara (X_c, Y_c, Z_c)

La transformación se realizará mediante una transformación euclídea según rotación y traslación como podemos observar en la siguiente figura extraída de [2].

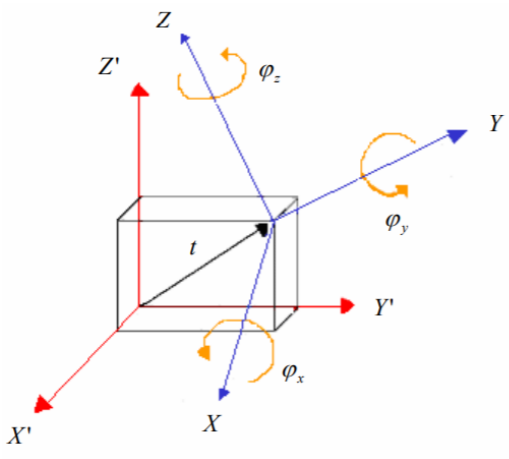

Figura 2.4: Transformación euclídea en 3D, extraído de [2].

En el cambio de coordenadas del mundo a las de la cámara intervienen la rotación y traslación:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \overrightarrow{t} \tag{2.2}$$

Utilizando las coordenadas homogéneas, con $\beta = 1$, y descomponiendo la matriz de rotación, R, en la multiplicación de matrices de rotación de cada eje obtenemos que $R = R_x \cdot R_y \cdot R_z$.

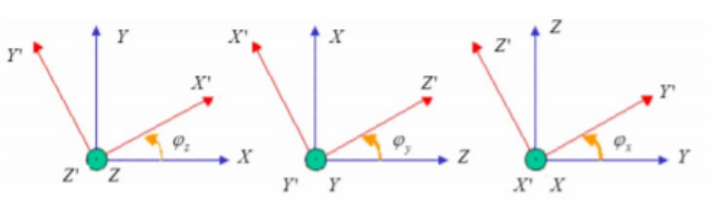

Figura 2.5: Rotación de los ejes, extraido de [2].

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\varphi_x & sin\varphi_x \\ 0 & -sin\varphi_x & cos\varphi_x \end{bmatrix} \tag{2.3}$$

$$R_y = \begin{bmatrix} cos\varphi_y & 0 & -sin\varphi_y \\ 0 & 1 & 0 \\ sin\varphi_y & 0 & cos\varphi_y \end{bmatrix} \tag{2.4}$$

$$R_z = \begin{bmatrix} cos\varphi_z & sin\varphi_z & 0 \\ -sin\varphi_z & cos\varphi_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

Una vez obtenidas las matriz de rotación de cada eje y la matriz R total, obtenemos el sistema de matrices intermedio dado por:

$$\begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{2.6}$$

que corresponde a la ecuación: $M_r = R \cdot M$.

Y finalmente sumando la traslación obtenemos la ecuación $M_c = M_r + t$ que se

descompone de la siguiente forma:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} X_r \\ Y_r \\ Z_r \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \tag{2.7}$$

Con todo ello ($\beta = 1$ y R) obtendremos:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{2.8}$$

Por último, la obtención de la matriz de proyección (de tamaño 3×4) se realiza a través de la matriz de parámetros intrínsecos, K, multiplicada por la matriz de rotación, R, extendida con la matriz de traslación, t:

$$P = \begin{bmatrix} f \cdot d_u & 0 & u_0 & 0 \\ 0 & f \cdot d_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.9}$$

con lo que tendríamos:

$$P = K \cdot [R|t]. \tag{2.10}$$

2.3.4. Matriz Fundamental

Una vez expuesto todo lo anterior llegamos al concepto de matriz fundamental, F. Se trata de la generalización de la matriz esencial que veremos a continuación en 2.3.5. Dicha matriz es, como se explica en [3], la representación matemática de la geometría epipolar puesto que relaciona puntos de una imagen m y m' con sus líneas epipolares, l y l'.

$$l' = F \cdot m \tag{2.11}$$

La matriz fundamental por tanto cumple la siguiente igualdad con la que se podrá calcular F teniendo las distintas correspondencias entre imágenes, m y m'

$$m' \cdot F \cdot m = 0 \tag{2.12}$$

Propiedades de la Matriz Fundamental

- Transposición: siendo F, en la ecuación 2.12 y 2.11, la matriz fundamental del par de cámaras P y P', F^T también lo será del par P y P'
- Lineas epipolares: para cualquier punto m en la primera imagen la línea epipolar viene dada por las ecuaciones 2.12 y 2.11. De igual manera, para el punto m', la relación será: $l = F^T \cdot m'$.
- Grados de libertad de la matriz: se trata de una matriz 3×3 con siete grados de libertad (siete términos independientes) y rango dos, lo que implica: $det(F) = 0$.

Estimación de la Matriz Fundamental

Basándonos en [8] podemos afirmar que conociendo las matrices de proyección P y P' seremos capaces de extraer las relaciones epipolares de la escena. Dichos parámetros en un principio son desconocidos y deben ser estimados a través de relaciones entre cámaras. Para la obtención de la geometría epipolar es necesario llevar a cabo algún método de estimación puesto que no puede ser estimada de forma directa.

Los métodos de estimación se basan en la resolución de ecuaciones deducidas a partir de 2.12:

$$U \cdot f = 0 \tag{2.13}$$

, donde

$$f = (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})^T \tag{2.14}$$

$$U = \begin{pmatrix} u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_nu'_n & u_nv'_n & u_n & v_nu'_n & v_nv'_n & v_n & u'_n & v'_n & 1 \end{pmatrix} \tag{2.15}$$

, donde $F_{i,j}$ es el elemento de la fila i y la columna j de la matriz fundamental y (u_n, v_n) y (u'_n, v'_n) son las coordenadas de los puntos correspondientes en los planos de imagen de cada una de las cámaras involucradas. En la matriz U aparecerán nueve incógnitas y siete términos independientes que vendrán determinados por las dos columnas independientes y el factor de escala. Todo esto hace que la matriz fundamental tenga rango dos.

Los tres métodos mas comunes para la estimación de la matriz fundamental son: métodos lineales, iterativos y robustos. Ofreceremos una visión general de estos pero nos centraremos en los métodos robustos.

- **Métodos lineales:** basado en el cálculo de la matriz fundamental usando siete correspondencias. La ventaja de este método es la poca información requerida para estimar la matriz fundamental. Por el contrario, este hecho provoca una estimación poco robusta ante errores y no puede ser utilizada en casos en los que haya mayor número de correspondencias de las necesarias.

- **Métodos iterativos:** pueden ser divididos en dos grupos: aquellos métodos en los que se minimiza la distancia entre los puntos y las líneas epipolares (haciendo mínima la ecuación del método, utilizando Newton-Raphson) y aquellos basados en gradiente.

- **Métodos robustos, [9]:** algunos ejemplos de estos métodos son: M-Estimators, Least-Median-Squares (LMedS), Random Sampling (RANSAC), MLESAC, [**?**], y MAPSAC. LMedS y RANSAC, explicado en [8, 10, 11], son técnicas muy similares. Ambas se basan en la selección aleatoria de conjuntos de puntos usados para conseguir una aproximación de la matriz fundamental, F, usando un método lineal. RANSAC calcula, para cada matriz, F, el número de *inliers* (mejor resultado) en el que la matriz es aquella que maximiza dicho número. Además, una vez se eliminan los *outliers* (peor resultado) la matriz, F, se calcula de nuevo para obtener una mejor estimación. La selección aleatoria de los puntos favorece la convergencia del proceso.

Medidas del error de proyección

Existen, principalmente, dos formas de medida del error, [12]: aquel calculado como el error cuadrático medio de la distancia entre correspondencias de la estimación, m y m', y la línea epipolar generada entre ellas mediante la ecuación 2.16 y aquel llamado error real en [12]. El error real consiste en crear un número n de correspondencias correctamente calculadas y hallar el error resultante mediante:

$$ErrorGeometrico(F) = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} dist(m, Fm')^2} \tag{2.16}$$

Posición relativa de una cámara respecto de la otra

Teniendo la estimación de la matriz fundamental continuaremos con la obtención de las matrices de proyección P y P'. Durante la transformación proyectiva del espacio 3D la matriz fundamental no varia ya que no depende de la elección del origen de coordenadas del mundo ni de los parámetros internos (calibración) de cada cámara.

Según [3], siendo H la transformación mediante geometría proyectiva del espacio 3D, la matriz fundamental para las matrices P y P' y las matrices PH y$P'H$ será idéntica.

El proceso de obtención de P quedaría de la siguiente forma:

- Definición de la matriz de proyección P de forma canónica, formada por la matriz identidad I de tamaño 3×3 extendida con un vector de ceros de tamaño 3×1

$$P = [I|0] \tag{2.17}$$

- Paralelamente definimos la matriz de proyección de la otra cámara, P', como:

$$P' = [(e^T)_x F_f | e^T] \tag{2.18}$$

siendo $(e')_x$ la expansión en una matriz antisimétrica del epipolo, e'.

- En cuanto al vector de traslación, $\overrightarrow{t}'$, y la matriz de rotación, R', del centro de proyección, C', respecto de C, podemos obtenerlo mediante la descomposición QR de la matriz ortogonal R y de la matriz de parámetros intrínsecos de la cámara K':

$$R' = \frac{P'(.,4) * (K')^{-1}}{||P'(.,4) * (K')^{-1}||_{fro}} \tag{2.19}$$

$$t' = -P'(.,1,2,3) \tag{2.20}$$

donde $P'(.,j)$ es la matriz que resulta de eliminar la columna j y el operador $||...||_{fro}$ obtiene la norma de Frobenius de una matriz cuadrada.

2.3.5. Matriz Esencial

La matriz esencial, E, permite la reconstrucción euclidiana de la escena a partir de las imágenes capturadas por un par de cámaras. Reúne la información necesaria relacionada con la geometría epipolar de dos puntos de vista con la calibración intrínseca de las cámaras. Haciendo uso de la matriz esencial podemos saber, dado un punto en una imagen, qué linea epipolar corresponde a un punto en cuestión en la otra imagen.

Basándonos en el material desarrollado en [13, 3, 14], para usar correctamente dicha matriz se asume que las cámaras satisfacen el modelo de cámaras *pinhole*, es decir, que conocemos las matrices de parámetros intrínsecos K (vista en la sección

2.3.1) para cada cámara involucrada y las matrices fundamentales F (vista en la sección 2.3.4) para cada par de cámaras. La matriz esencial se define como:

$$E = R \cdot [t]_x \tag{2.21}$$

donde $[t]_x$ es el producto vectorial de los vectores de t. Aplicando este concepto y la propiedad de la matriz de rotación, $R^T \cdot R = I$, llegamos a la ecuación de Longuet-Higgins [13] con la que obtendremos la matriz esencial:

$$E = K_1^T \cdot F \cdot K_2 \tag{2.22}$$

Propiedades de la matriz esencial

- Si es multiplicada por un escalar no nulo el resultado es otra matriz esencial con las mismas características e información que la matriz inicial. Esto hace que la matriz se incluya dentro del concepto de espacio proyectivo. Este concepto se relaciona con la forma en la que las cámaras capturan las escenas (transforman el 3D a 2D). Un punto que se encuentra en una linea de proyección se proyecta en un punto de imagen común. Por tanto, se trata de una matriz invariante a escalado.

- Es de tamaño 3×3. Tiene 5 grados de libertad y rango 2, es decir, $det(E) = 0$. La matriz de rotación, R, tiene 3 grados de libertad al igual que la matriz de traslación, t. Considerando que la matriz esencial es invariante a escalado y siendo un elemento proyectivo deberemos restar un grado de libertad a los 6 grados que suman las matrices de las que se extrae.

2.3.6. *Bundle Adjustment*

Según Hartley y Zisserman en [3], es el método por excelencia para el problema de la reconstrucción de escenas 3D. Dicho método iterativo no asegura la convergencia de la solución, es decir, puede que no obtengamos una solución válida o lo suficientemente buena para la situación en cuestión. La ventaja del uso de *bundle adjustment* es que es un algoritmo general que puede ser aplicado a un gran espectro de situaciones en los que haya problemas de reconstrucción y optimización.

Pongamos un caso práctico, extraído de [3], para explicar dicho proceso, en el que, traduciendo el nombre del método, ajustamos el haz de luz capturado por las cámaras:

Consideramos un escenario en el que hay un conjunto de puntos 3D definidos por

X_j y capturados por un conjunto de cámaras con matrices P conocidas (matriz de características vista en 2.3.3). El problema a resolver es: dado el conjunto de imágenes x_j^i(coordenadas del punto j obtenido por la cámara i) hallar el conjunto de matrices de proyección P y los puntos X_j que hacen que se cumpla la igualdad:

$$P^i \cdot X_j = x_j^i \tag{2.23}$$

Si las medidas obtenidas de la imagen, X_j, son ruidosas, la ecuación anterior no podrá ser perfectamente satisfecha. En dicha situación buscaremos la solución del Máximo Vecindario o Maximum Likelihood (ML) asumiendo que el ruido introducido en los puntos X_j es gaussiano. Estimaremos las matrices de proyección, P^i, y los puntos 3D, X_j, que se proyecten exactamente en los puntos de la imagen y minimizaremos la distancia entre puntos reproyectados y medidos para cada situación en la que el punto 3D aparece. Dicha estimación en la que se minimiza el error de proyección es el conocido como *bundle adjustment*. Dicho de otra forma, ajustamos el haz de luz entre el centro de cada cámara, $C^1, C^2, ..., C^i$ en 2.3, y el conjunto de puntos 3D (así como entre cada punto 3D y el conjunto de centros de cámaras).

Este método iterativo es utilizado generalmente como paso final para cualquier reconstrucción. Su gran ventaja es la flexibilidad ante la falta de datos gracias a la solución del Máximo Vecindario. Seria un método perfecto salvo por el hecho de que requiere una buena inicialización y porque puede llegar a ser un gran problema debido al gran número de parámetros involucrados en dicho algoritmo (requiere mucha información inicial para hacer uso del algoritmo en cuestión). Alguna posible solución para ello sería reducir el número de cámaras o el número de puntos X_j que proporcionamos al algoritmo, dividir el conjunto de datos en varios grupos de menor tamaño y después volver a agregarlos como al inicio...

2.4. Detección, descripción y correspondencias de puntos de interés

Los puntos de interés, como veremos a continuación, son puntos o regiones características de la imagen. Existen detectores, en el que se decide que región puede ser considerada como punto representativo de la imagen, y descriptores que reúnen las características de aquellos puntos detectados.

2.4.1. ¿Que es un punto de interés?

Se trata de las características de un punto que hacen que se pueda diferenciar del resto de puntos de una imagen. En otras palabras, se trata de un descriptor local.

Siguiendo lo expuesto en [15], estos puntos de interés (*Points of Interest* o PoI), que aparecen normalmente en las esquinas de la intersección de dos o más bordes de imágenes, están claramente definidos en el espacio (sus coordenadas son conocidas), proporcionan mucha información sobre el contenido de la imagen y son muy estables en cuanto a cambios locales o globales en la imagen. Estas variaciones en la imagen, como explica [16], se deben principalmente al escalado de la imagen, rotaciones, cambios de perspectiva, traslaciones de la misma o cambios en la iluminación.

Los puntos de interés son usados como la característica local en la mayoría de aplicaciones de recuperación de imagen basada en el contenido o el reconocimiento de objetos. Además, pueden ser buenos indicadores de contornos de objetos y situaciones de oclusión de objetos en secuencias de imágenes.

Algunos de los métodos de extracción de puntos de interés más conocidos son el algoritmo de Moravec [17], el operador SUSAN [18], Harris y Stephens [19] *Genetic-Programming* [20], SIFT [21] y SURF [22] entre otros.

2.4.2. Métodos de detección

El objetivo de los métodos de detección es la localización robusta y precisa de puntos invariantes a escalados, cambios de perspectiva, traslaciones de la misma, cambios en la iluminación o rotaciones.

Scale-Invariant Feature Transform (SIFT)

Una de las cualidades mas útiles del detector de características SIFT, en [21], es que es robusto ante un nivel de ruido de píxel alto. El mayor motivo de error es la localización inicial y la detección de la escala de la imagen. SIFT es tanto descriptor como detector de puntos de interés.

Un breve resumen del proceso de detección mediante SIFT es:

1. Detección de *scale-space*, [23]: esta etapa de búsqueda se realiza en cada una de las escalas y ubicaciones de la imagen. Se implementa de manera eficiente utilizando la diferencia de Gaussianas (DoG) para obtener los máximos y mínimos relativos en un volumen dado. Estos puntos de interés detectados son invariantes a escala y a orientación.

2. Localización de puntos de interés: se seleccionan en función de su estabilidad. Se crea un modelo que determina la escala y localización para cada candidato a ser punto de interés.

3. Asignación de orientación: para cada punto detectado se asignan una o mas orientaciones. Las operaciones futuras realizadas sobre cada uno de los puntos tendrán que realizarse teniendo en cuenta la orientación asignada, la escala y la localización proporcionando así invarianza a las transformaciones.

4. Descriptor: el gradiente de la imagen local es medido en la escala elegida en la región cercana al punto de interés. Así podemos transformarlos en representaciones que permitan niveles apreciables de distorsiones en la forma y cambios en la iluminación.

***Speeded Up Robust Features* (SURF)**

Algoritmo inspirado en SIFT que detecta la región cercana a un punto de interés, [22]. Es una variación del método anterior en el que se utiliza también la diferencia de Gaussianas (DoG) junto a un banco de filtros de caja con tamaño variable que simula el efecto *scale-space*. Así conseguimos un algoritmo más rápido y eficiente computacionalmente.

En cuanto a los inconvenientes de ambos métodos podemos incluir los problemas que se presentan a la hora de la detección en zonas de poca textura (zonas planas), la necesidad de extraer mucha información antes de poder utilizar el método y el hecho de que dos implementaciones distintas del algoritmo puede que no den los mismos resultados.

2.4.3. Métodos de descripción

Basado en la caracterización de una región de la imagen. El descriptor también sera invariante a escalados, cambios de perspectiva, traslaciones de la misma, cambios en la iluminación o rotaciones. Dichos descriptores podrán ser aplicados a puntos de interés o a todos los píxeles de una imagen (descripción densa de la imagen).

SIFT y SURF, además de ser métodos de descripción, también son descriptores, [21, 24]. SIFT se inspira en modelos biológicos del sistema visual humano en el que se relaciona las distribuciones de gradiente con la capacidad humana de reconocer objetos. Consta de dos etapas: una inicial en la que se asigna la orientación de cada punto, como podemos ver en la Figura 2.6 extraída de [21], y otra en la que se elige una región de tamaño 16×16 en torno al punto clave y se obtiene para cada píxel

que lo compone el valor de escala y orientación del gradiente. Este proceso otorga al descriptor de robustez a desplazamientos localizados, a cambios en la iluminación y a la cuantificación de las orientaciones.

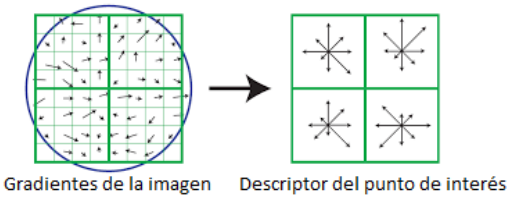

Figura 2.6: Orientaciones del descriptor SIFT

El descriptor SURF, [24], se basa, de nuevo, en propiedades similares a SIFT. Soluciona el problema de la orientación basándose en la información presente en un área circular alrededor del punto de interés y, finalmente, construye una región cuadrada que se alinea con la orientación seleccionada para extraer el descriptor SURF definitivo. El descriptor, por tanto, será similar al de la Figura 2.6.

Estos descriptores tienen el inconveniente de que solo serán invariantes a pequeñas variaciones del punto de vista (como cambios de perspectiva, cambios de iluminación o rotaciones leves) por motivos de diseño del mismo.

2.4.4. Método de obtención de correspondencias

En esta sección abordaremos el método de extracción de correspondencias con el que obtenemos la menor distancia entre descriptores, m y m'.

Previamente a la obtención de correspondencias deberemos haber hecho la detección y la descripción de los puntos de interés, 2.4.1.

Estableceremos dichas correspondencias entre puntos de interés mediante un vector de características. Para ello, como se explica en [21], basta con calcular la distancia euclídea entre los puntos de interés de ambas imágenes, A y B, utilizando:

$$d(m, m') = \sqrt{\sum(\varphi(m) - \varphi(m'))^2} \tag{2.24}$$

en la que el operador φ representa el parámetro donde se extrae el descriptor del punto en la imagen A o B.

Para un punto m de la imagen A su descriptor será aquel punto m' contenido en el plano de imagen B cuya distancia sea mínima. Por tanto, el descriptor m' será el más parecido a m.

En el caso de correspondencias ruidosas o distorsionadas se seleccionará aquellas

correspondencias η veces menor que la obtenida para cualquier otro punto de la imagen a la que corresponda dicho descriptor. En este caso η se fija por parte del usuario como umbral a partir del cuál se hace la decisión, siendo este más restrictivo cuanto mayor sea dicho umbral.

2.5. Segmentación de regiones

Para esta sección diferenciaremos entre borde, contorno y región. Un contorno delimita un objeto y su región es el área contenido en dicho contorno. Por el contrario, el borde delimita un objeto pero no tiene porque ser cerrado. A continuación veremos también una explicación sobre el método de segmentación de regiones conocido como *Ultrametric Contour Map* (UCM).

2.5.1. *Ultrametric Contour Map*

La detección de contornos, como podemos extraer de [25, 26, 27], es una decisión binaria puesto que se tratará de un contorno o no independientemente del objeto. Es la decisión mas sencilla a la hora de segmentar imágenes aunque sigue suponiendo un reto. Por ello se propone el uso de *Ultrametric Contour Map* (UCM). El nivel básico del método propone la segmentación más exhaustiva de la imagen (segmentar mas de lo necesario) creando mayor numero de regiones de menor tamaño como podemos observar en la figura 2.7b. Los niveles superiores del método respetan solo los contornos claramente definidos. Este nivel provocara una segmentación opuesta al nivel básico con menor número de regiones (Figura 2.7c y 2.7d). Utilizando lo mejor de ambos extremos obtendremos la mejor segmentación posible.

El método consta de varios pasos:

1. Definición de los elementos que intervendrán en el proceso. Se define un grafo inicial $G = (\Theta_0, \Delta_0, W(\Delta_0))$ en el que los nodos (vértices) son las regiones Θ_0, los arcos que los unen son Δ_0 y los pesos de cada nodo vienen definidos por $W(\Delta_0)$, que son una medida de la diferencia entre regiones.

2. El algoritmo ordena los arcos, Δ_0, según similitud y mediante un método iterativo une aquellas regiones mas similares, o lo que es lo mismo, selecciona los menores pesos para una región y agrega regiones con mayor similitud a la región seleccionada. La similitud o no entre dos regiones Θ_0 adyacentes viene dada por el valor medio del arco común Δ_0 cuyo valor sera el peso definido por $W(\Delta_0)$. El método se repite hasta no poder realizar la asociación entre regiones. Ese momento llega cuando los contornos restantes tienen pesos mayores o iguales

a las regiones anteriormente agregadas entre si, por tanto, el peso de dichas regiones no decrecerá.

3. El proceso genera un árbol de regiones en el que la raíz es la imagen inicial y las hojas son los elementos de Θ_0. Se encuentran ordenadas según el proceso iterativo anteriormente descrito.

El resultado del proceso, la imagen U, puede representarse mediante un dendrograma (gráfico en forma de árbol que organiza datos en categorías) donde el valor $H(\Theta_0)$ de cada región Θ_0 es la diferencia respecto al valor de su primera aparición en el método.

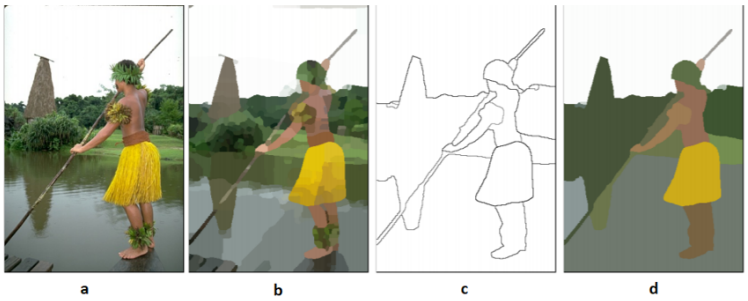

Figura 2.7: Segmentación jerárquica de contornos mediante UCM.
Extraído de [26]. La imagen **a** es la inicial y **b** su segmentación correspondiente al nivel mas exhaustivo de UCM, con regiones representadas por su media de color. Por último, **c** es la imagen de contornos y **d** su correspondiente segmentación fijando el umbral a 0.5, con lo que obtenemos una segmentación con menor numero de regiones.

Capítulo 3

Diseño y Desarrollo

3.1. Introducción

En este capítulo describimos el procedimiento realizado para obtener la calibración extrínseca y reconstrucción 3D de una escena capturada por varias cámaras. La escena está iluminada (además de por luz natural) por un patrón predeterminado, que varía temporalmente, emitido por un proyector. El proceso se estructura en diferentes módulos, organizados como en la Figura 3.1. El capítulo comienza con el diseño del patrón emitido y posteriormente describe cada uno de los módulos de la Figura 3.1.

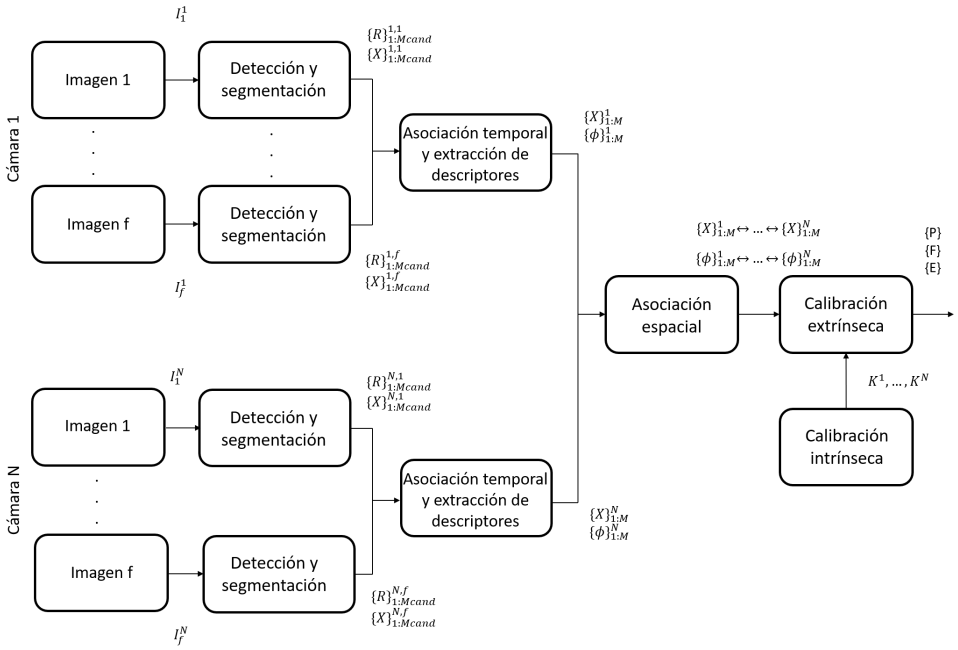

Figura 3.1: Diagrama de bloques del proceso

Primero, capturamos la escena formada por el patrón proyectado y el escenario creado. Dicha imagen, I_f^N, donde f es el número de *frames* de cada secuencia y N el numero de cámaras que intervienen, se somete a un proceso de segmentación y detección en el cuál se extraen los centros, $\{X\}_{1:Mcand}^{N,f}$, y las regiones, $\{R\}_{1:Mcand}^{N,f}$, candidatas a ser descriptores. A continuación se realiza la extracción de descriptores y su asociación temporal con el que obtendremos el valor RGB de cada región, es decir, el descriptor $\{\Phi\}_{1:M}^{N}$ (donde N es el número de cámaras y M el número de descriptores) y su centro $\{X\}_{1:M}^{N}$. Una vez hecha la asociación temporal pasamos a su asociación espacial para cada par de imágenes dando como resultado un conjunto de correspondencias entre cámaras para cada *frame* procesado ($\{X\}_{1:M}^{1} \longleftrightarrow ... \longleftrightarrow \{X\}_{1:M}^{N}$). Por último, haciendo uso de la calibración intrínseca de las cámaras y de la información extraída hasta el momento conseguimos obtener las matrices de calibración extrínseca P, F y E, como veremos en la sección 3.6.

3.2. Diseño del patrón emitido

En esta fase inicial se crea una escena para grabar una secuencia de imágenes como vemos en la Figura 3.2. El concepto es crear un escenario en el que se proyecta una imagen sobre un plano sin reflejos y en el cuál tenemos un control sobre la luz del espacio de grabación. Dicha luz ambiente será generalmente de baja intensidad para maximizar la visibilidad del patrón proyectado.

El patrón emitido tiene información similar pero no igual en cada *frame*. La idea es poder detectar como varía ese punto temporalmente. Para ello, cada punto tiene que tener una secuencia en tiempo que lo describa totalmente distinta al punto contiguo.

En este trabajo se han utilizado cinco cámaras para la captura de una escena estática puesto que, aunque el número de cámaras se puede aumentar, permite la reconstrucción fiable de la escena. Como salida de este módulo tenemos el conjunto de imágenes, I_f^N.

También debemos mencionar que para la extracción de la matriz Esencial, es necesaria la calibración intrínseca de todas las cámaras que intervienen en el proceso. La calibración intrínseca se realiza capturando, con cada cámara, distintas imágenes de un damero y procesándolas mediante la aplicación *Camera Calibration* proporcionada por Matlab.

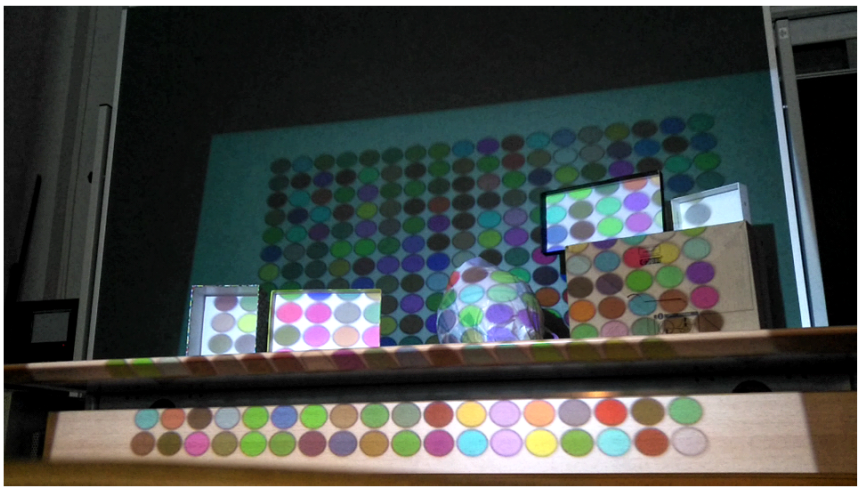

Figura 3.2: Diseño del patrón emitido

3.3. Detección y segmentación

En este primer bloque, haciendo uso del algoritmo *Ultrametric Contour Map* (expuesto en 2.5.1) y utilizando imágenes preprocesadas, obtendremos la detección de contornos (Figura 3.4a), y, por tanto, la segmentación en regiones de la secuencia grabada. Este método será más exhaustivo o menos dependiendo del valor umbral fijado en el mismo.

Preprocesado de imágenes

Consiste en modificar la imagen capturada con el objetivo de mejorar el resultado del algoritmo UCM. En este caso se ha aumentando el contraste.

Ultrametric Contour Map (UCM)

A cada imagen preprocesada se le aplica este algoritmo con el que obtenemos la imagen de regiones (ver Figura 3.4a). Realizaremos la detección para todas las cámaras involucradas y para todas las imágenes capturadas.

Segmentación en regiones

Extraemos las características asociadas a cada región dada por la imagen de contornos U. Esto nos permitirá obtener el color RGB, el centro y la etiqueta (que nos sirve para identificar de manera sencilla cada región dentro de una imagen) para

posteriormente crear el descriptor. El umbral utilizado, th_1, en este caso para la detección ha sido fijado a 0.145. Con este valor haremos la segmentación en regiones.

El color RGB de cada región se obtiene asignando a cada etiqueta el valor medio RGB de todos los píxeles que la componen. A continuación se crea la imagen de medias que, finalmente, definirá cada región. Por último, extraeremos las coordenadas del centro de cada región para ser capaces posteriormente de hacer la asociación. Dichos centros se han obtenido con la transformada de la distancia para cada una de las etiquetas.

De este primer módulo obtenemos los candidatos a ser descriptores. Tendremos, por tanto, la información de las regiones candidatas, $\{R\}^{N,f}_{1:Mcand}$, que contiene la información del color RGB y etiquetas de cada región, y los centros candidatos, $\{X\}^{N,f}_{1:Mcand}$. El resultado se puede observar gráficamente en la Figura 3.4b.

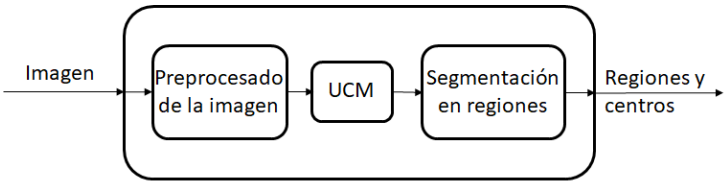

Figura 3.3: Detalle módulo de detección y segmentación

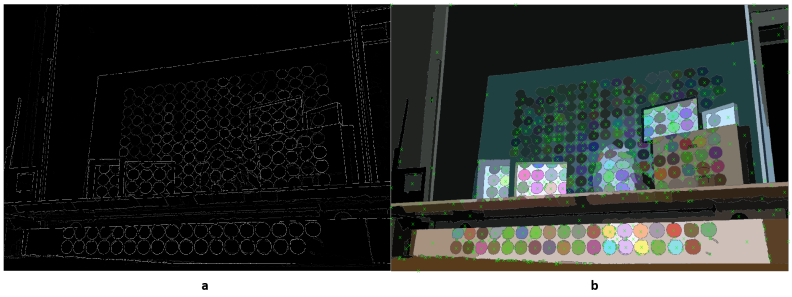

Figura 3.4: Detección y segmentación. **a.** Detección de contornos mediante UCM. **b.** Extracción de centros y regiones con su respectivo color RGB.

3.4. Asociación temporal y extracción de descriptores

Haciendo uso de la información obtenida en el apartado anterior planteamos la extracción de descriptores y su posterior asociación temporal.

Las regiones cumplen dos propiedades:

1. Tiempo: Una región idéntica a la anterior, en cuanto a su localización, no tiene el mismo valor RGB en ningún otro *frame*. Esta propiedad se basa en la diferencia entre cada *frame* proyectado en un vídeo.

2. Espacio: En un mismo *frame* no hay dos colores iguales, o, en otras palabras, no hay dos valores RGB exactamente iguales que definan una región.

Extracción del descriptor

Para la extracción de descriptores utilizamos la característica que diferencia cada región. Los descriptores candidatos, $\{\varphi\}^{N,f}_{1:Mcand}$, vendrán dados por el color RGB contenido en los valores de las regiones candidatas $\{R\}^{N,f}_{1:Mcand}$. Dichos descriptores permiten que cada región sea perfectamente distinguible en tiempo y en espacio.

Asociación temporal

Siguiendo la propiedad de tiempo anteriormente expuesta, realizamos la asociación temporal, en la Figura 3.6a, para la duración integra de la secuencia de *frames*. Tomamos como referencia la imagen inicial para asociar cada región del *frame* N, que está siendo procesado, con el inicial. De esta manera la asociación se hace por parejas: I^1_1 con I^1_2, I^1_1 con I^1_3, ... ,I^1_1 con I^1_f hasta obtener una asociación temporal de todas las imágenes proyectadas. Para ello asumimos estaticidad en la captura y estabilidad en la segmentación de regiones. Asociamos una región del *frame* f con una del *frame* 1 si su centro pertenece al área definida por la región en el *frame* inicial. Tras este proceso obtendremos los descriptores de cada cámara N, $\{\Phi\}^N_{1:M^*}$, y centros, $\{X\}^N_{1:M^*}$, que hemos sido capaces de asociar temporalmente y que posteriormente serán filtrados.

Refinado de las asociaciones

Esta asociación temporal tiene errores, por tanto, en este bloque también se hace un refinado de los datos extraídos del proceso. Para eliminar las regiones distintas a los patrones proyectados que forman parte del fondo o parte de la escena no alcanzada por el patrón se eliminan aquellos descriptores, $\{\Phi\}^N_{M^*}$, cuya varianza en tiempo sea menor que un umbral determinado. Por otro lado, para eliminar regiones cuya

segmentación es inestable, se eliminan aquellos descriptores de regiones que no se asocien temporalmente un mínimo de veces a la región inicial. En este caso el valor de la varianza mínima, th_2, y el porcentaje de asociaciones mínima, th_2, se ha fijado a 40 % y 0.002 respectivamente. Un ejemplo del filtrado puede observarse atendiendo a los centros, $\{X\}_M^N$, representados en la Figura 3.6b.

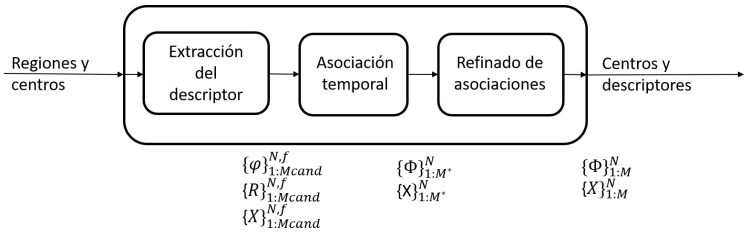

Figura 3.5: Detalle asociación temporal y refinado.

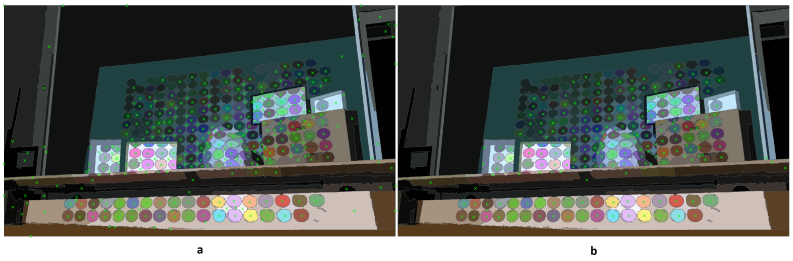

Figura 3.6: Asociación temporal y refinado

3.5. Asociación espacial

Este proceso, basado en la propiedad de espacio descrita en la sección 3.4, consiste en obtener la correspondencia entre regiones de cada cámara utilizando los descriptores temporales agregados.

Para cada *frame* tendremos la información de descripción de la región, $\{\Phi\}_{1:M}^N$, (en forma de valores RGB) y de los centros, $\{X\}_{1:M}^N$, del tamaño del número de *frames* de la secuencia procesada. Este descriptor se utilizará para relacionar espacialmente cada una de las regiones del patrón proyectado según cada cámara.. Así uniremos la información obtenida durante la asociación temporal a la información que se extrae al hacer la asociación espacial. Este proceso da como resultado un conjunto de correspondencias entre cámaras, $\{X\}_{1:M}^1 \longleftrightarrow ... \longleftrightarrow \{X\}_{1:M}^N$, como podemos ver en la Figura 3.7.

Para realizar la asociación espacial se ha hecho uso de una función disponible en

el laboratorio de investigación. Dicha función recibe como parámetros los centros, $\{X\}_{1:M}^{N}$, y los descriptores, $\{\Phi\}_{1:M}^{N}$, extraídos anteriormente. Hace uso de la distancia SSD (Sum of Squared Differences) para medir la distancia entre los vectores de características de la primera cámara respecto de los de la segunda. Así obtenemos la información de las correspondencias espaciales para un número de cámaras N.

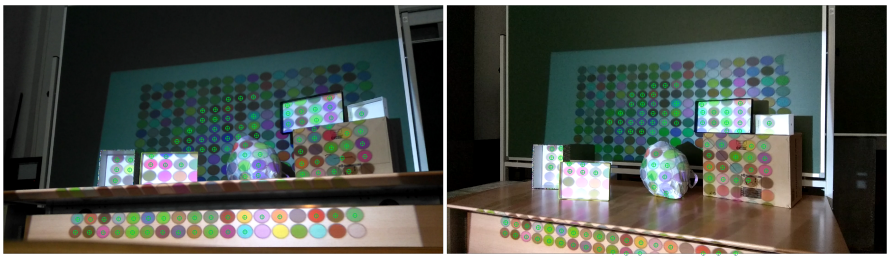

Figura 3.7: Asociación espacial, nótese la diferencia respecto a la Figura 3.6.

3.6. Calibración extrínseca

Una vez obtenidas las correspondencias entre cámaras, haremos uso de ellas para calcular las matrices que definen la calibración extrínseca (posición relativa entre cada par de cámaras). Esta calibración extrínseca puede hacerse métrica utilizando la calibración intrínseca de cada cámara.

A partir de las correspondencias obtenidas, seleccionamos dos cámaras. Mediante RANSAC (expuesto en 2.3.4) estimaremos la matriz fundamental, F_{xy} , siendo x e y la cámara seleccionada. Una vez aplicado RANSAC podemos estimar las matrices de proyección, P, de las cámaras implicadas en el proceso, [3]. Utilizando la información 2D de las correspondencias y la información de las matrices de proyección, para el par de cámaras analizados, podremos estimar los puntos 3D por triangulación. Para el conjunto de cámaras restante utilizaremos la información de los puntos 3D obtenidos de este proceso y las correspondencias 2D en cada cámara para calcular el resto de matrices de proyección. Así habremos calculado las matrices de proyección del par de cámaras elegido por estimación y el del resto de pares de cámaras gracias a los puntos 2D y a los puntos 3D obtenidos durante la estimación de las matrices de proyección de las cámaras seleccionadas.

Después de estos procesos aplicamos el método de corrección de errores *Bundle Adjustment*. En la Figura 3.8a podemos observar que el error de reproyección es alto puesto que hay valores del histograma superiores a 200 píxeles en valor absoluto. Gracias al método aplicado conseguimos reducir dicho error considerablemente, ver

Figura 3.8b, puesto que los errores elevados desaparecen y se centran, la mayoría, alrededor del cero.

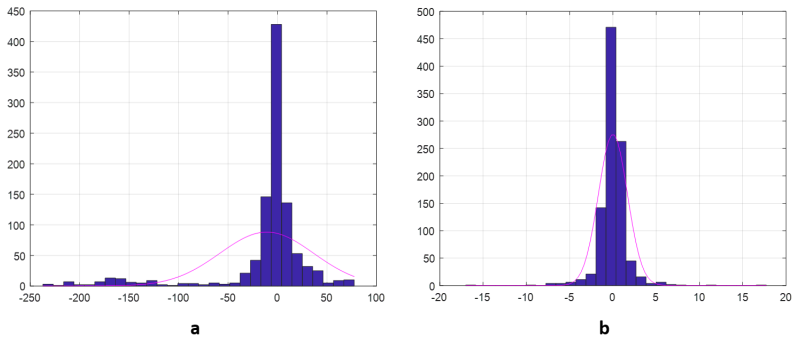

Figura 3.8: Corrección de error de reproyección aplicando *Bundle Adjustment*. **a**. Error de reproyección antes de aplicar el algoritmo. **b**. Error de reproyección después de aplicarlo.

3.7. Calibración intrínseca

Cada cámara tendrá unas características propias y por ello es importante la calibración de todos los dispositivos de grabación que intervengan. Se ha utilizado la aplicación *Camera Calibrator* proporcionada por Matlab para su calibración (Figura 3.9). Su uso consiste en introducir un conjunto de imágenes de un damero (elemento comúnmente usado para la calibración) capturadas con cada una de las cámaras que han intervenido en el proceso de grabación del patrón inicial.

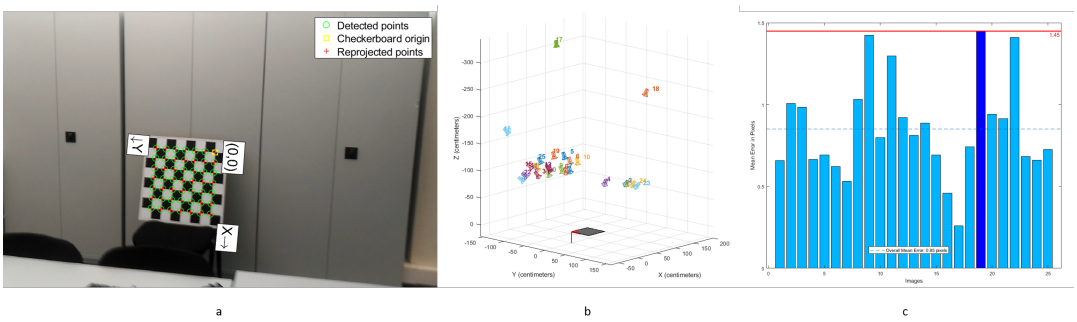

Figura 3.9: Calibración intrínseca de la cámara

Capítulo 4

Resultados experimentales

4.1. Introducción

En este capítulo pondremos en relieve los resultados obtenidos en el trabajo. Presentaremos varias pruebas de los distintos métodos y procesos utilizados aportando tanto datos cualitativos como cuantitativos de los mismos. También haremos una breve discusión sobre ellos.

4.2. Marco de evaluación

Presentaremos a continuación las pruebas realizadas durante la realización de este trabajo. Introduciremos la base de datos de imágenes, el código y el software utilizado en cada caso. Además, compararemos y analizaremos los métodos que han sido usados durante la realización del proceso descrito en el Capitulo 3.

Para la realización de estas pruebas hemos utilizado una secuencia de un patrón emitido cualquiera. La calibración extrínseca de la escena puede ser mas fácil o mas difícil dependiendo del escenario recreado.

En cuanto a los valores tenidos en cuenta para realizar las pruebas hemos hecho variar aquellos parámetros que permitían mejorar el resultado final para ver su impacto en la calibración de las cámaras.

4.3. Pruebas experimentales

A continuación mostramos una serie de pruebas realizadas sobre el algoritmo cuyo objetivo es la calibración de las cámaras. Realizamos pruebas sobre la calibración intrínseca de las cámaras, la detección de puntos de interés y la asociación temporal.

4.3.1. Calibración intrínseca de las cámaras

Extraemos los valores de la calibración de las cámaras que han intervenido en la grabación de la escena usando el software proporcionado por Matlab. Para los valores obtenidos, en la Tabla 4.1, podemos observar que para las cinco cámaras el valor del error reproyección medio de todas las cámaras es muy bajo, consiguiendo así información fiable de los parámetros intrínsecos de la cámara como lo son el centro óptico, c, de la misma y el factor de escalado, k_u y k_v, como hemos expuesto en 2.3.1.

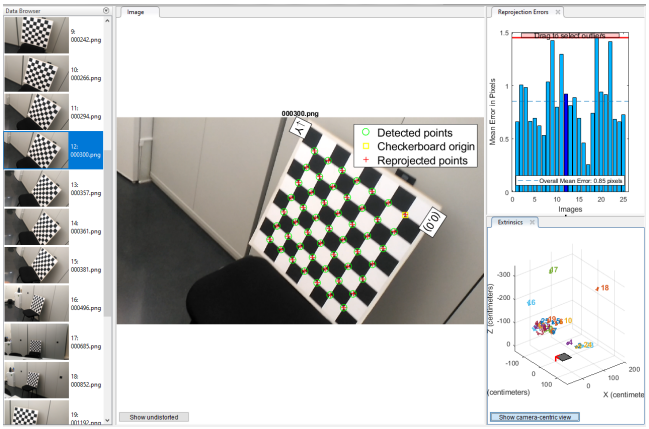

Figura 4.1: Interfaz para la calibración intrínseca de una de las cámaras. Error de reproyección medio y situación relativa de las cámaras a la hora de la calibración.

Calibración intrínseca	Número de imágenes de calibración	Error de reproyección medio (pixels)
Cámara 1	25	0,85
Cámara 2	21	1,57
Cámara 3	25	1,82
Cámara 4	15	1,67
Cámara 5	33	1,44

Tabla 4.1: Datos de calibración de la red de cámaras

4.3.2. Detección de puntos de interés y asociación temporal

A continuación compararemos los distintos valores y umbrales definidos durante la realización del trabajo. Para ello haremos la representación, una vez hecha la detección mediante el algoritmo UCM, visto en 2.5.1, la extracción de puntos de interés del resultado obtenido y su posterior asociación temporal, de los puntos detectados por cada cámara para cada uno de los valores de la prueba realizada. Hemos hecho variar el valor de umbral th_1 para el algoritmo UCM entre 0.12 y 0.16, observando una gran variación en la detección de regiones y la posterior extracción de descriptores, puesto

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,12	30	0,001	250	321	341	262	330
		0,0015	238	301	298	244	308
		0,002	229	291	275	238	294
		0,0025	222	286	264	229	285
		0,003	212	280	260	223	282

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,12	40	0,001	281	346	370	280	373
		0,0015	265	325	321	258	346
		0,002	254	313	297	249	326
		0,0025	245	305	285	239	312
		0,003	231	299	281	231	307

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,12	50	0,001	295	357	382	294	395
		0,0015	277	335	331	268	366
		0,002	265	323	306	258	342
		0,0025	256	314	294	246	327
		0,003	241	308	289	238	321

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,13	30	0,001	246	303	322	246	324
		0,0015	238	285	283	231	304
		0,002	225	276	265	226	290
		0,0025	222	274	253	218	283
		0,003	210	268	248	211	281

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,13	40	0,001	275	332	355	268	358
		0,0015	263	309	307	249	334
		0,002	248	297	286	242	313
		0,0025	242	295	274	234	303
		0,003	225	288	267	224	299

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,13	50	0,001	282	343	369	282	376
		0,0015	269	319	319	259	352
		0,002	254	306	298	250	329
		0,0025	248	304	285	240	317
		0,003	231	296	277	229	313

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,1375	30	0,001	240	292	302	235	308
		0,0015	232	277	263	224	290
		0,002	225	270	246	215	281
		0,0025	218	263	239	211	273
		0,003	208	261	235	203	271

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,1375	40	0,001	269	325	348	263	345
		0,0015	259	303	297	250	325
		0,002	251	294	278	239	308
		0,0025	241	285	270	234	297
		0,003	224	281	264	222	294

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,1375	50	0,001	277	334	363	272	360
		0,0015	266	311	312	254	340
		0,002	258	302	292	242	321
		0,0025	248	293	282	236	310
		0,003	231	289	276	224	307

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,1475	30	0,001	230	283	286	227	301
		0,0015	223	272	259	214	283
		0,002	217	265	239	208	276
		0,0025	208	259	233	205	266
		0,003	203	258	231	198	264

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,1475	40	0,001	257	312	329	259	338
		0,0015	247	297	296	243	317
		0,002	239	287	273	233	306
		0,0025	228	278	265	230	294
		0,003	216	275	263	218	291

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,1475	50	0,001	269	323	344	264	350
		0,0015	258	306	309	247	328
		0,002	249	296	284	237	315
		0,0025	237	287	275	234	302
		0,003	224	282	273	222	299

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,15	30	0,001	224	280	285	224	300
		0,0015	218	267	258	215	280
		0,002	213	262	239	209	275
		0,0025	204	256	231	205	266
		0,003	199	254	228	198	263

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,15	40	0,001	252	312	327	251	330
		0,0015	243	296	295	240	307
		0,002	238	288	272	230	299
		0,0025	225	279	262	226	288
		0,003	215	275	259	215	284

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,15	50	0,001	267	324	338	258	350
		0,0015	257	306	305	245	323
		0,002	250	298	281	235	312
		0,0025	234	287	270	231	300
		0,003	223	282	267	220	296

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,16	30	0,001	215	272	267	222	284
		0,0015	209	261	244	215	275
		0,002	203	257	229	205	264
		0,0025	193	251	221	199	260
		0,003	187	248	217	196	256

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,16	40	0,001	244	301	317	242	312
		0,0015	236	287	288	231	299
		0,002	229	283	268	219	286
		0,0025	216	273	256	212	281
		0,003	207	268	252	207	277

Umbral UCM	Umbral Numero Asociaciones (%)	Umbral Varianza	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
0,16	50	0,001	260	311	330	256	329
		0,0015	252	296	299	242	314
		0,002	243	291	277	228	297
		0,0025	225	280	265	221	291
		0,003	215	273	260	216	286

Tabla 4.2: Pruebas realizadas para distintos valores del umbral de detección, th_1, y del posterior refinado de descriptores, th_2 y th_3. Para cada combinación de parámetros se indica el número de regiones detectadas en cada cámara tras la asociación espacial (ver sección 3.5). El número de regiones proyectadas en el patrón es 326.

que al aumentar el nivel de detección solo se respetaban los contornos claramente definidos (niveles superiores del algoritmo). Se ha llegado a un compromiso en el que visualmente y de manera subjetiva la detección era la óptima. Además, como se representa en la Figura 4.2, se ha hecho un barrido para la limpieza realizada, th_2, después de la asociación temporal entre 0.001 y 0.003 para el valor de la varianza mínima, th_3, que debe tener cada descriptor para ser tratado como tal y entre 30 % y 50 % para el número de veces que un descriptor se asocia correctamente en tiempo con el descriptor de las posteriores imágenes.

Los resultados obtenidos han sido que para un patrón emitido la obtención óptima de regiones del patrón emitido ha sido para el valor de UCM de 0.12, 50 % para el número mínimo de asociaciones temporales y un valor de varianza de 0.001. Los dos últimos valores proporcionan, a la hora del refinado, el mejor resultado posible puesto que aportan el mayor número de descriptores finales. Esto, aunque no sea una

medida de calidad del método utilizado, proporciona la mejor información posible para la posterior asociación espacial. Se ha elegido la detección que se acerca mas al numero de regiones a detectar del patrón emitido.

4.3.3. Matriz fundamental

Comprobamos los valores del error de reproyección de los distintos pares de matrices fundamentales ($F_{12}, F_{13}, ..., F_{45}$). Mostramos el error de reproyección (medido en pixeles) y los valores después de aplicar el método de *Bundle Adjustment*, en las Tablas 4.3 y 4.4. Podemos observar como, aplicando el método de corrección de errores, el error de reproyección mejora en gran medida. Las Tablas mostradas son matrices simétricas y por ello no se muestra información redundante. La matriz $F_{11}, ..., F_{55}$,no tiene sentido alguno puesto que estaríamos comparando la misma cámara (por ello la diagonal principal de ambas Tablas no se muestra). Se puede ver que para algún par de cámaras de la Tabla 4.4 los valores de error de la matriz fundamental son bastante bajos (en F_{13}), con lo que podríamos obtener una reconstrucción de la imagen robusta a errores y que representa bastante fielmente la realidad.

En la Figura 4.2 podemos observar el GUI (Interfaz gráfica de usuario) en el que comprobamos que la matriz fundamental esta correctamente calculada puesto que la linea epipolar pasa por el punto exacto que hemos seleccionado en la imagen contigua.

Matriz F	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
Cámara 1	-	-	-	-	-
Cámara 2	479,02	-	-	-	-
Cámara 3	3,64	21,57	-	-	-
Cámara 4	9,46	109,50	267,20	-	-
Cámara 5	222,22	117,38	706,15	313,55	-

Tabla 4.3: Valores de error de reproyección de la matriz fundamental para todas las cámaras (en pixels). La resolución de las cámaras es 1080x1920.

Bundle Adjustment	Cámara 1	Cámara 2	Cámara 3	Cámara 4	Cámara 5
Cámara 1	-	-	-	-	-
Cámara 2	64,68	-	-	-	-
Cámara 3	2,39	9,73	-	-	-
Cámara 4	5,13	68,52	127,24	-	-
Cámara 5	36,30	52,06	182,55	107,42	-

Tabla 4.4: Valores de error de reproyección después de aplicar *Bundle Adjustment* (en pixels). La resolución de las cámaras es 1080x1920.

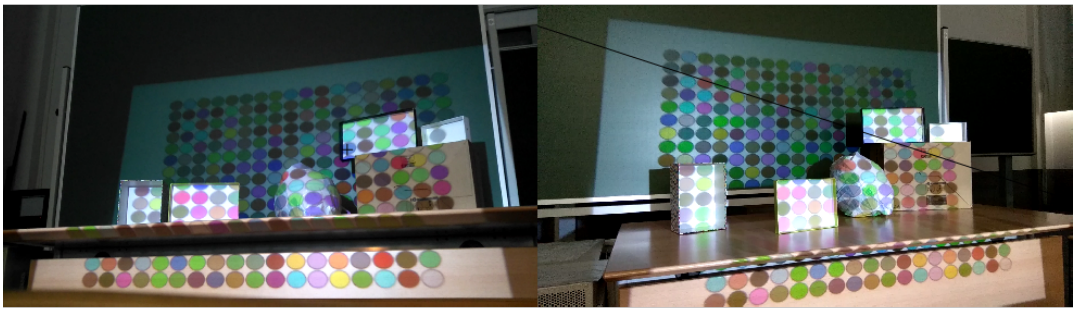

Figura 4.2: GUI matriz fundamental

4.3.4. Representación de los puntos 3D

Se trata de la representación en 3D de los puntos asociados espacial y temporalmente finales. En la Figura 4.4 podemos observar los puntos que obteníamos como resultado de la asociación espacial en la Figura 4.3.

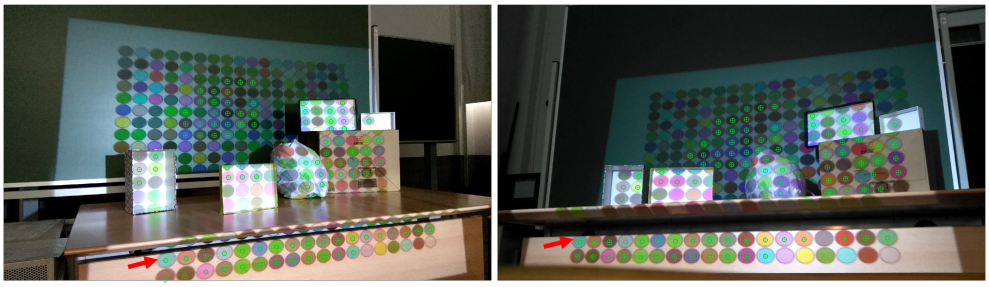

Figura 4.3: Asociaciones espaciales

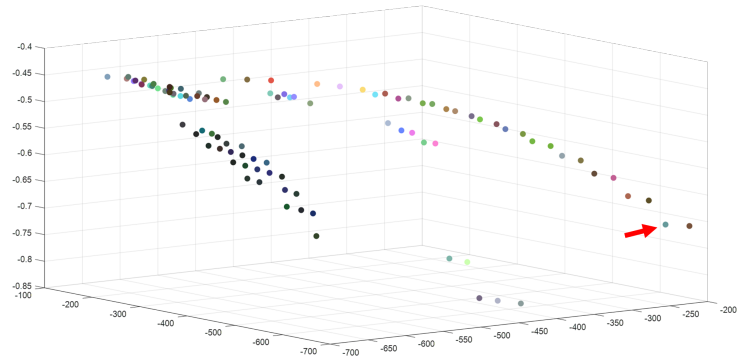

Figura 4.4: Representación 3D de los puntos asociados espacialmente

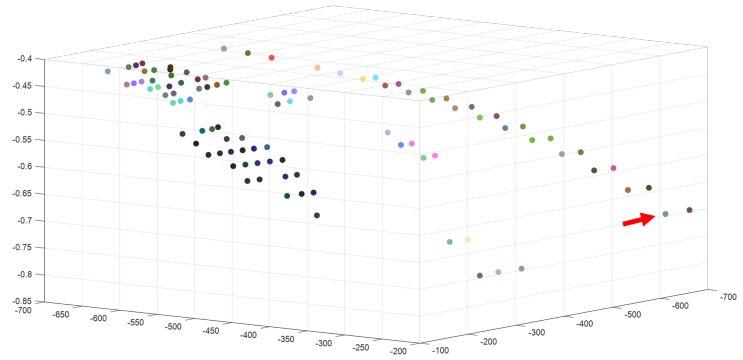

Figura 4.5: Representación métrica de los puntos asociados espacialmente

4.4. Discusión

Habiendo expuesto estas prueba representativas del trabajo realizado podemos decir que la calibración intrínseca de las cámaras se realiza con gran precisión gracias al uso de la aplicación de Matlab. En cuanto a la asociación temporal hay varios valores que pueden mejorar la calidad de la detección y posterior asociación. Esto dependerá de si la imagen consta de muchos contornos o no, puesto que la detección se deberá ajustar a las condiciones del escenario para la detección óptima.

Para las matrices fundamentales podemos afirmar, observando las Tablas proporcionadas en la Sección 4.3.3, que para al menos un par de cámaras la matriz fundamental y por tanto la matriz de proyección retorna errores de retro-proyección moderados, especialmente despúes de aplicar el método *Bundle Adjustment*.

Capítulo 5

Conclusiones y trabajo futuro

5.1. Conclusiones

Este método, basándonos en [1], ofrece una alternativa robusta ante errores puesto que se tiene en consideración información tanto espacial como temporal de la secuencia de imágenes capturada. En el caso del estado del arte, se ha conseguido profundizar en la geometría proyectiva comprendiendo el funcionamiento del paso del 3D de la realidad a las imágenes 2D capturadas por las cámaras.y el concepto de parámetros extrínsecos e intrínsecos que permiten la posterior calibración de las cámaras.

Durante el desarrollo del algoritmo se ha comprendido de manera gráfica como se realiza la calibración extrínseca de la escena y cómo, para ello, se extraen los distintos parámetros y características relacionadas con la auto calibración de una red de cámaras.

Además se han realizado pruebas para los distintos parámetros que podemos hacer variar durante el método. Hemos obtenido la mejor segmentación posible de forma subjetiva y la mejor limpieza posible de descriptores para dicho valor fijado en el algoritmo de detección y segmentación. Para las matrices fundamentales hemos obtenido valores aceptables en cuanto error de reproyección. Así demostramos que el método es prometedor y que es posible obtener valores de error asumibles.

5.2. Trabajo futuro

A la vista de los resultados que se han obtenido en este trabajo se propone trabajar en la calibración de una red de cámaras con un patrón no estático y la creación de una interfaz para la representación 3D con la información extraída por el método presentado en este trabajo. Esta interfaz podría hacer uso de la herramienta COLMAP

para la reconstrucción con texturas de la escena que no ha podido ser obtenida en este TFG como se pretendía. También se propone la creación de un escenario más complejo aumentando el número de cámaras y creando una escena con mayor número de objetos a reconstruir. Además sería interesante crear un método de decisión para la detección de segmentación en el algoritmo UCM, una forma en la que no intervenga la opinión del programador, una forma en la que medir objetivamente la efectividad de una detección y segmentación en regiones.

Bibliografía

[1] K. Ide, S. Siering, and T. Sikora, "Automating multi-camera self-calibration," in *2009 Workshop on Applications of Computer Vision (WACV)*, pp. 1–6, IEEE, 2009.

[2] J. Garcia, "Autocalibración y sincronización de múltiples cámaras ptz," *PFC in UAM*, 2007.

[3] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[4] J. Smisek, M. Jancosek, and T. Pajdla, "3d with kinect," in *Consumer depth cameras for computer vision*, pp. 3–25, Springer, 2013.

[5] Q.-T. Luong and O. D. Faugeras, "Camera calibration, scene motion and structure recovery from point correspondences and fundamental matrices," *Ijcv*, vol. 22, no. 3, pp. 261–289, 1997.

[6] O. Faugeras, *Three-dimensional computer vision: a geometric viewpoint*. MIT press, 1993.

[7] Q.-T. Luong and O. D. Faugeras, "The fundamental matrix: Theory, algorithms, and stability analysis," *International journal of computer vision*, vol. 17, no. 1, pp. 43–75, 1996.

[8] X. Armangué and J. Salvi, "Overall view regarding fundamental matrix estimation," *Image and vision computing*, vol. 21, no. 2, pp. 205–220, 2003.

[9] P. H. Torr and D. W. Murray, "The development and comparison of robust methods for estimating the fundamental matrix," *International journal of computer vision*, vol. 24, no. 3, pp. 271–300, 1997.

[10] R. C. Bolles and M. A. Fischler, "A ransac-based approach to model fitting and its application to finding cylinders in range data.," in *IJCAI*, vol. 1981, pp. 637–643, 1981.

[11] P. H. Torr and D. W. Murray, "The development and comparison of robust methods for estimating the fundamental matrix," *International journal of computer vision*, vol. 24, no. 3, pp. 271–300, 1997.

[12] F. Espuny and P. Monasse, "Singular vector methods for fundamental matrix computation," in *Pacific-Rim Symposium on Image and Video Technology*, pp. 290–301, Springer, 2013.

[13] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, no. 5828, p. 133, 1981.

[14] Z. Zhang, "Estimating motion and structure from correspondences of line segments between two perspective images," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 17, no. 12, pp. 1129–1139, 1995.

[15] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International journal of computer vision*, vol. 60, no. 1, pp. 63–86, 2004.

[16] R. Criado, M. Romance, and A. Sanchez, "Interest point detection in images using complex network analysis," *Journal of Computational and Applied Mathematics*, vol. 236, no. 12, pp. 2975–2980, 2012.

[17] H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover.," tech. rep., STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1980.

[18] H. Wu, J. Inada, T. Shioyama, Q. Chen, and T. Simada, "Automatic facial feature points detection with susan operator," in *Proceedings of the Scandinavian Conference on Image Analysis*, pp. 257–263, 2001.

[19] I. Pratikakis, M. Spagnuolo, T. Theoharis, and R. Veltkamp, "A robust 3d interest points detector based on harris operator," in *Eurographics workshop on 3D object retrieval*, vol. 5, Citeseer, 2010.

[20] L. Trujillo and G. Olague, "Automated design of image operators that detect interest points," *Evolutionary Computation*, vol. 16, no. 4, pp. 483–507, 2008.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[23] T. Lindeberg, *Scale-space theory in computer vision*, vol. 256. Springer Science & Business Media, 2013.

[24] P. Panchal, S. Panchal, and S. Shah, "A comparison of sift and surf," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 1, no. 2, pp. 323–327, 2013.

[25] P. Arbelaez, "Boundary extraction in natural images using ultrametric contour maps," in *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pp. 182–182, IEEE, 2006.

[26] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 5, pp. 898–916, 2011.

[27] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "From contours to regions: An empirical evaluation," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2294–2301, IEEE, 2009.